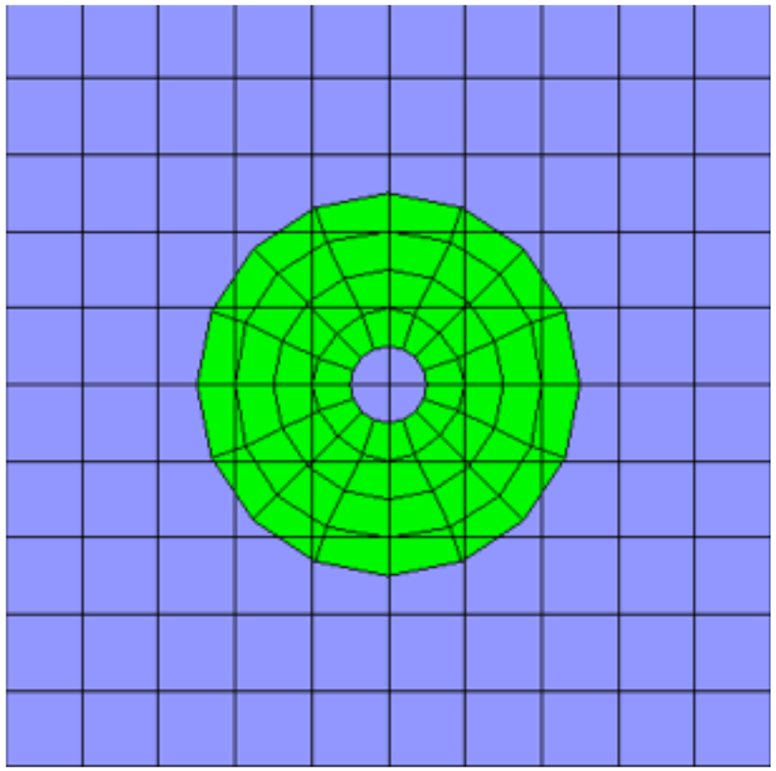
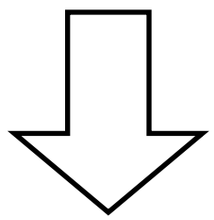
Chimera type

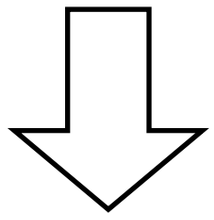
Domain Composition Methods

What?



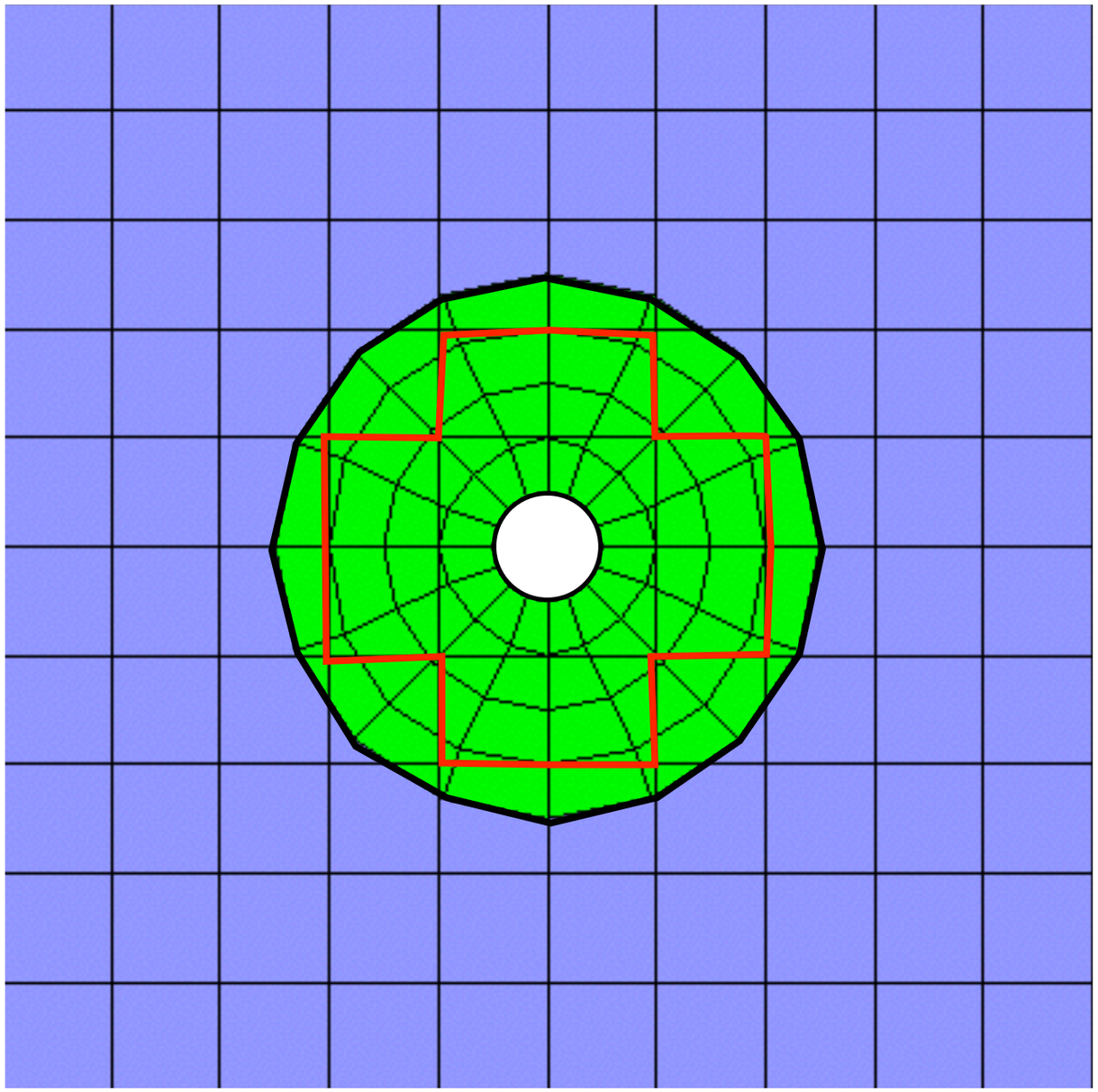


Hole cutting

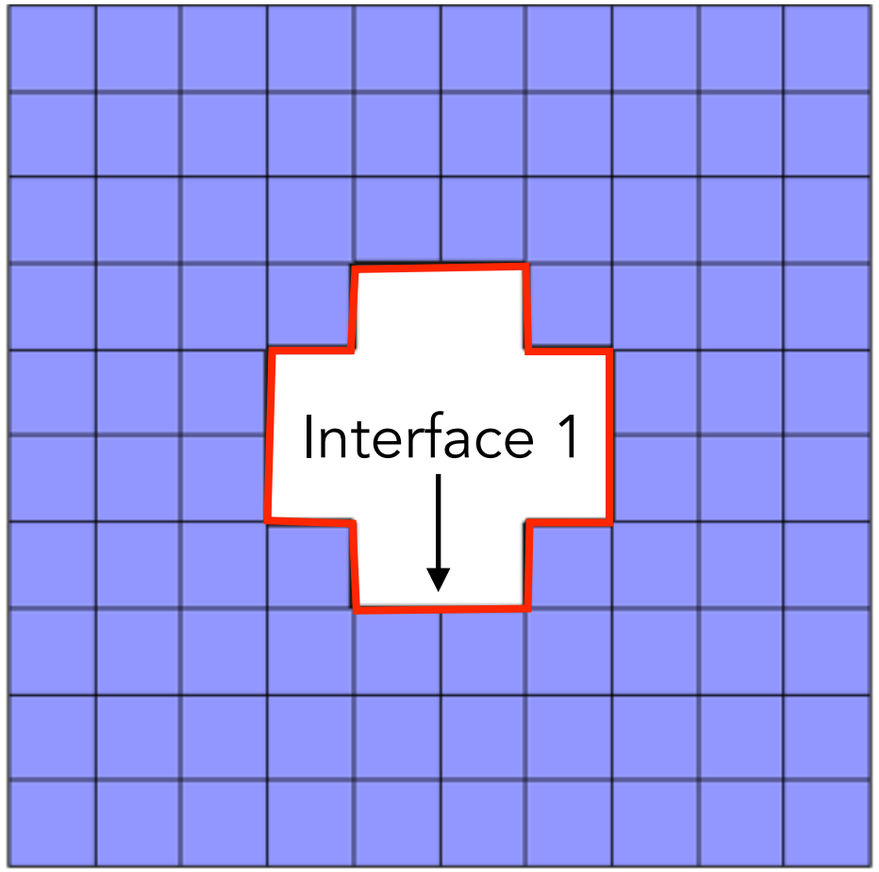


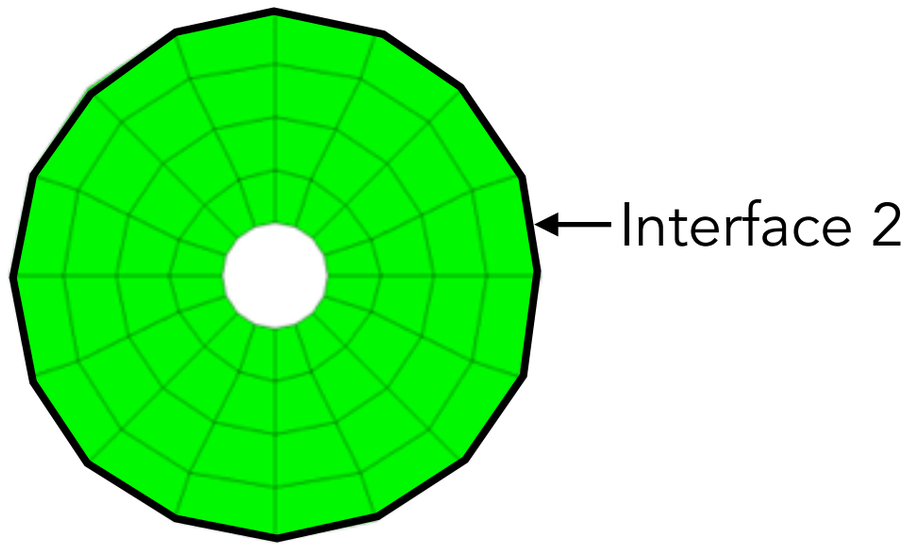
Couple meshes

What?



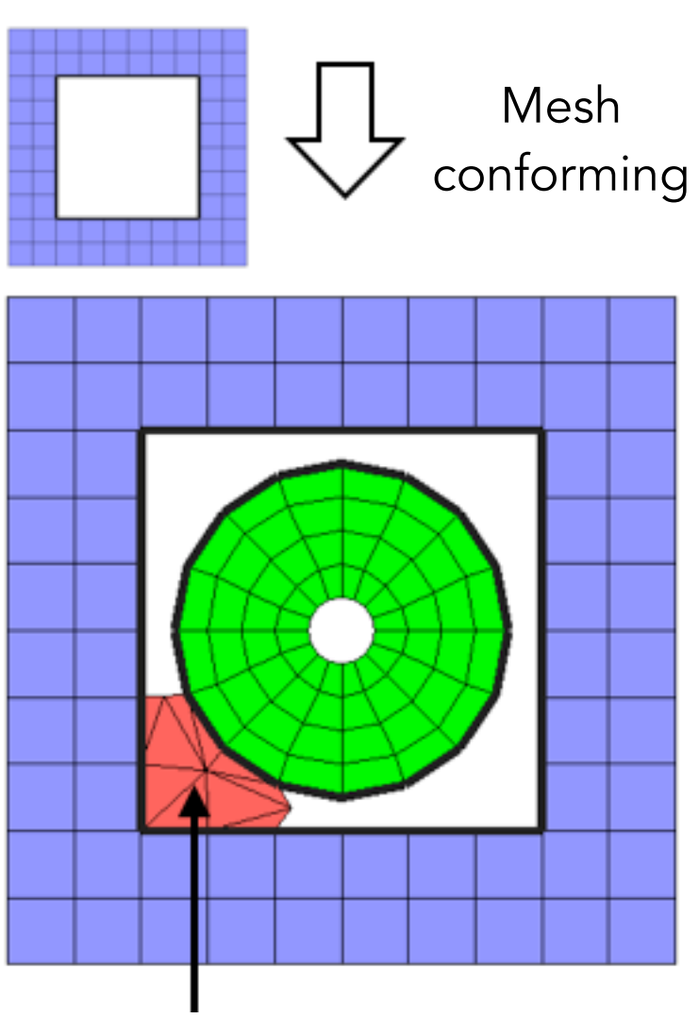
Chimera overlapped problems





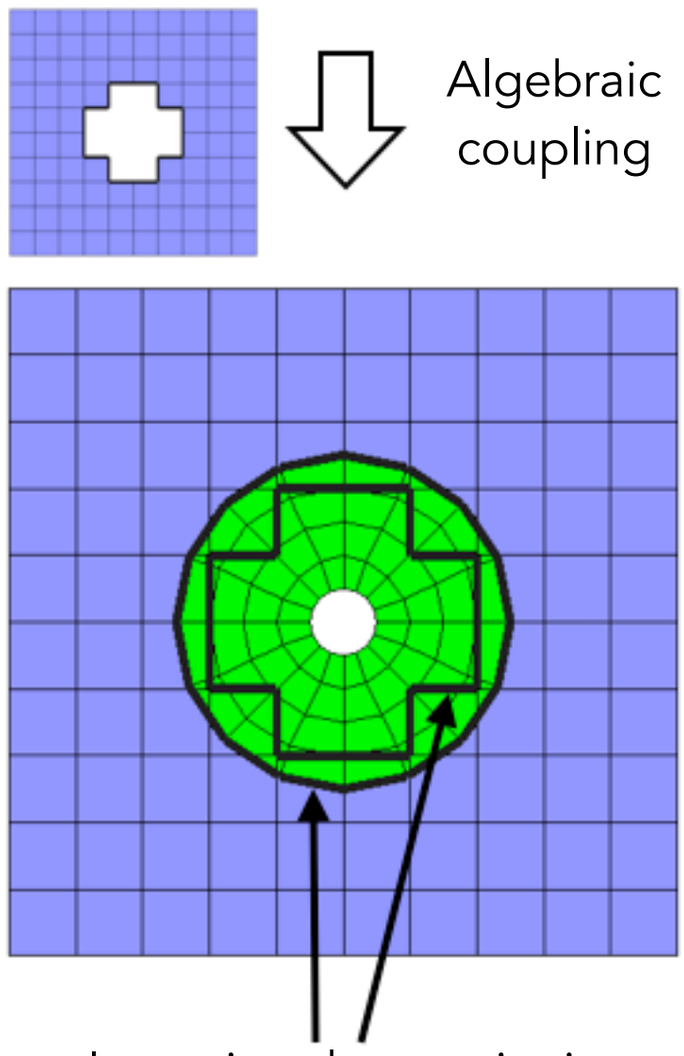
Hole cutting

What?



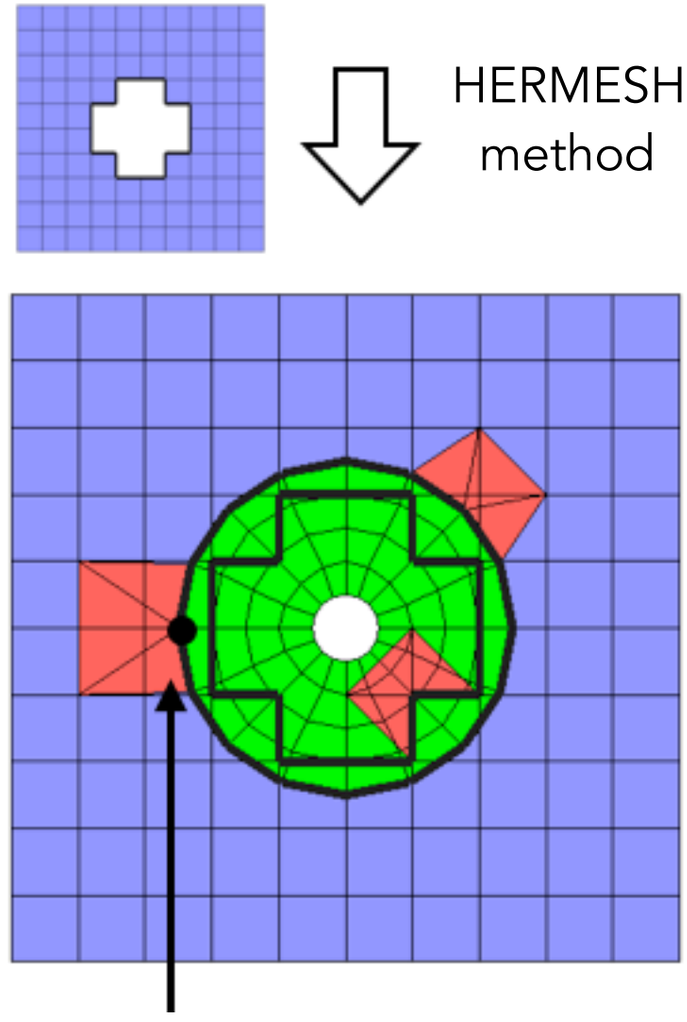
Mesh conforming

Example of elements to fill in the gap



Algebraic coupling

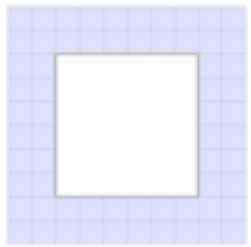
Imposing the continuity of the unknown and the residual



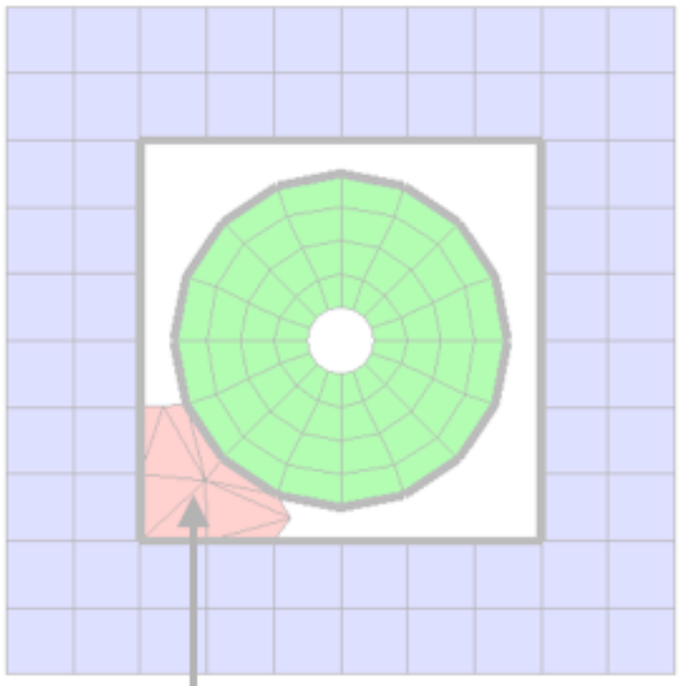
HERMESH method

Example of extension element for node ●

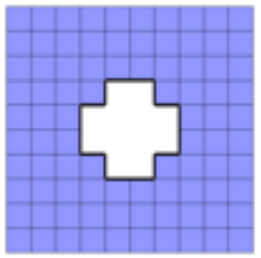
What?



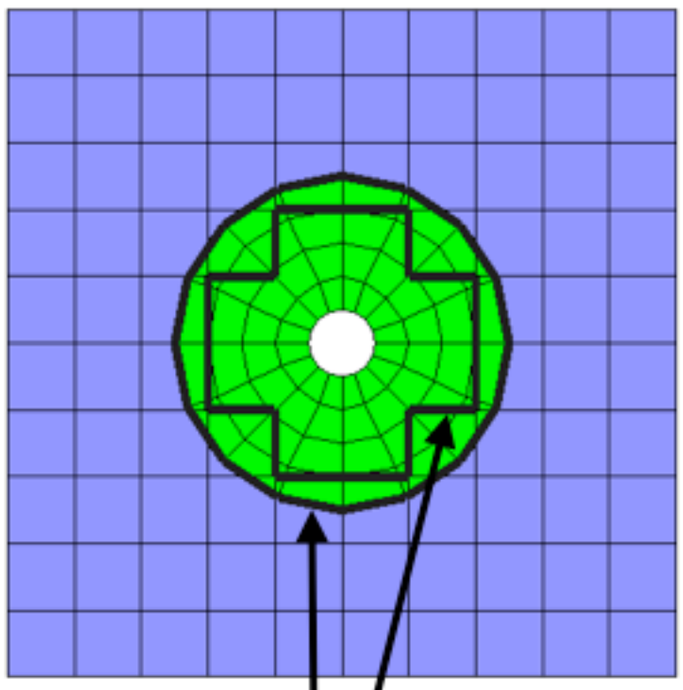
Mesh conforming



Example of elements to fill in the gap

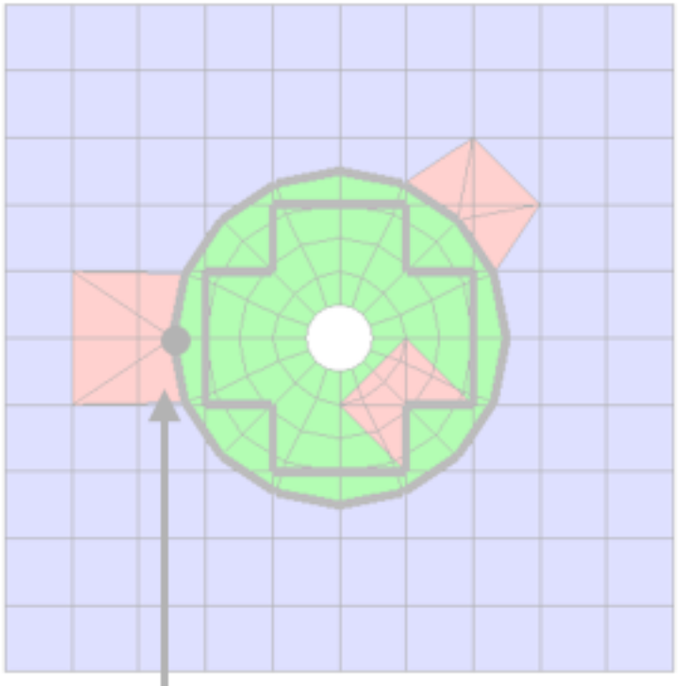


Algebraic coupling



Imposing the continuity of the unknown and the residual

HERMESH method



Example of extension element for node ●

Why?

CAD component union

Simplify meshing

Add/remove components

Moving objects

No remeshing in time

Add/remove components

Optimisation

Change location of components without having to remesh

Local mesh refinement

Overset a fine patch mesh

Enhanced collaborations

Meshes coming from different sources (biomechanical applications)

Why these couplings?

Versatile

It should be valid for any partial differential equation

Implicit

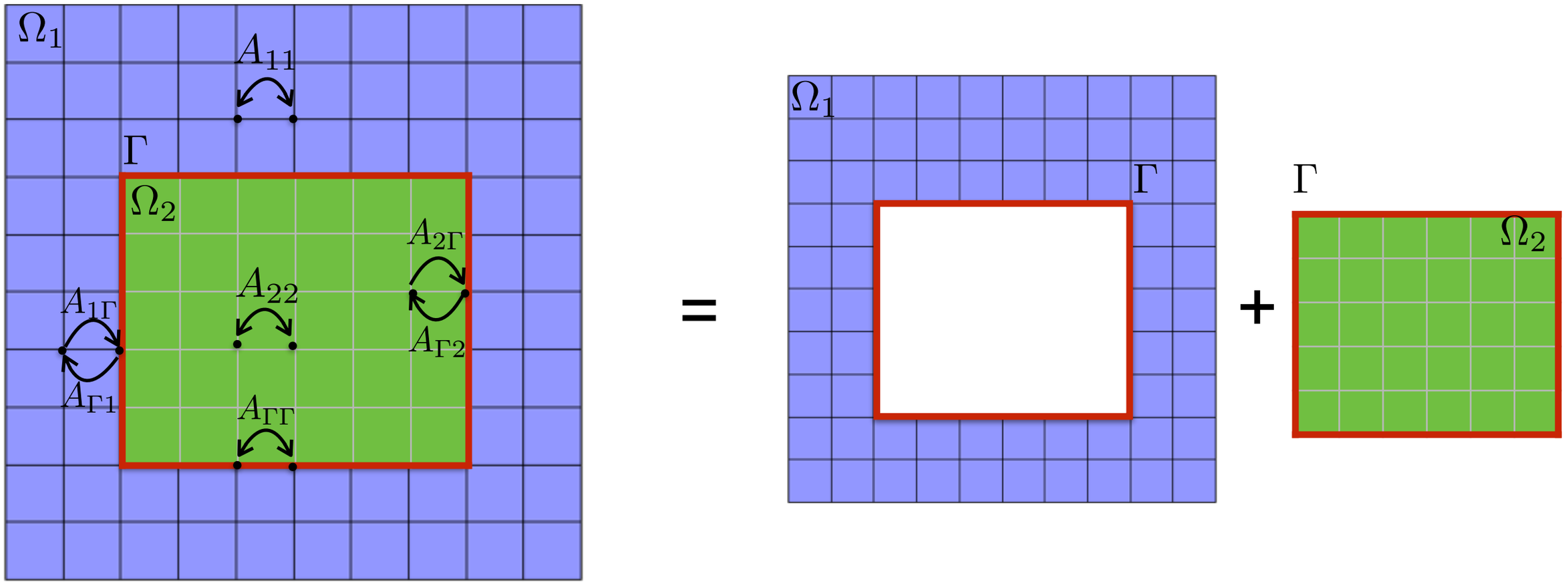
Does not introduce additional iterative loop

Parallel

It should be a parallel method

How? *ALGEBRAIC:* matrix-vector product level

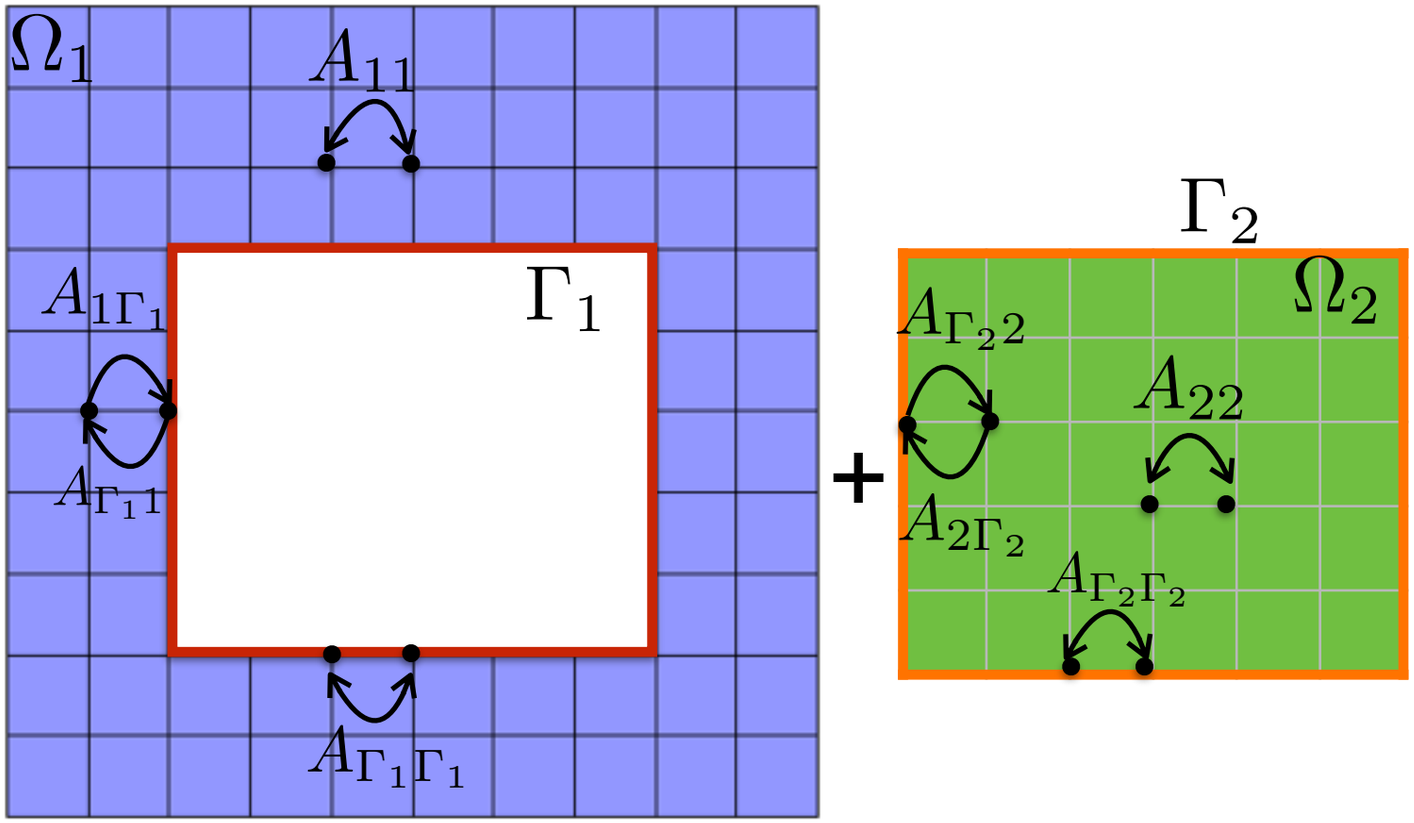
$$Au = b$$



$$\begin{pmatrix} A_{11} & 0 & A_{1\Gamma} \\ 0 & A_{22} & A_{2\Gamma} \\ A_{\Gamma 1} & A_{\Gamma 2} & A_{\Gamma\Gamma} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_\Gamma \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_\Gamma \end{pmatrix}$$

How? *ALGEBRAIC:* matrix-vector product level

Nodes duplicated on the interface



$$\Leftrightarrow$$

$$\begin{pmatrix} A_{11} & A_{1\Gamma_1} & 0 & 0 \\ A_{\Gamma_1 1} & A_{\Gamma_1\Gamma_1} & 0 & 0 \\ 0 & 0 & A_{22} & A_{2\Gamma_2} \\ 0 & 0 & A_{\Gamma_2 2} & A_{\Gamma_2\Gamma_2} \end{pmatrix} \begin{pmatrix} u_1 \\ u_{\Gamma_1} \\ u_2 \\ u_{\Gamma_2} \end{pmatrix} = \begin{pmatrix} b_1 \\ b_{\Gamma_1} \\ b_2 \\ b_{\Gamma_2} \end{pmatrix}$$

No-coupled system

How? *ALGEBRAIC:* matrix-vector product level

No-coupled system

$$\begin{pmatrix} A_{11} & A_{1\Gamma_1} & 0 & 0 \\ A_{\Gamma_1 1} & A_{\Gamma_1\Gamma_1} & 0 & 0 \\ 0 & 0 & A_{22} & A_{2\Gamma_2} \\ 0 & 0 & A_{\Gamma_2 2} & A_{\Gamma_2\Gamma_2} \end{pmatrix} \begin{pmatrix} u_1 \\ u_{\Gamma_1} \\ u_2 \\ u_{\Gamma_2} \end{pmatrix} = \begin{pmatrix} b_1 \\ b_{\Gamma_1} \\ b_2 \\ b_{\Gamma_2} \end{pmatrix}$$

$$r_{\Gamma_i} = b_{\Gamma_i} - A_{\Gamma_i i} u_i - A_{\Gamma_i \Gamma_i} u_{\Gamma_i}$$

Residual of the interface

How? *ALGEBRAIC*: matrix-vector product level

$$\begin{pmatrix} A_{11} & A_{1\Gamma_1} & 0 & 0 \\ 0 & I & 0 & -I \\ 0 & 0 & A_{22} & A_{2\Gamma_2} \\ 0 & 0 & A_{\Gamma_2 2} & A_{\Gamma_2\Gamma_2} \end{pmatrix} \begin{pmatrix} u_1 \\ u_{\Gamma_1} \\ u_2 \\ u_{\Gamma_2} \end{pmatrix} - \begin{pmatrix} b_1 \\ 0 \\ b_2 \\ b_{\Gamma_2} + r_{\Gamma_1} \end{pmatrix}$$

← Dirichlet

← Neumann

How? *ALGEBRAIC*: matrix-vector product level

$$\begin{pmatrix} A_{11} & A_{1\Gamma_1} & 0 & 0 \\ 0 & I & 0 & -I \\ 0 & 0 & A_{22} & A_{2\Gamma_2} \\ 0 & 0 & A_{\Gamma_2 2} & A_{\Gamma_2\Gamma_2} \end{pmatrix} \begin{pmatrix} u_1 \\ u_{\Gamma_1} \\ u_2 \\ u_{\Gamma_2} \end{pmatrix} = \begin{pmatrix} b_1 \\ 0 \\ b_2 \\ b_{\Gamma_2} + r_{\Gamma_1} \end{pmatrix} \leftarrow \text{Dirichlet} \quad \leftarrow \text{Neumann}$$

$$\begin{pmatrix} A_{11} & A_{1\Gamma_1} & 0 & 0 \\ 0 & I & 0 & -I \\ 0 & 0 & A_{22} & A_{2\Gamma_2} \\ 0 & 0 & A_{\Gamma_2 2} & A_{\Gamma_2\Gamma_2} \end{pmatrix} \begin{pmatrix} u_1 \\ u_{\Gamma_1} \\ u_2 \\ u_{\Gamma_2} \end{pmatrix} = \begin{pmatrix} b_1 \\ 0 \\ b_2 \\ b_{\Gamma_2} + r_{\Gamma_1} + D_2(u_{\Gamma_1} - u_{\Gamma_2}) \end{pmatrix} \leftarrow \text{Dirichlet} \quad \leftarrow \text{Robin}$$

How? *ALGEBRAIC*: matrix-vector product level

$$\begin{pmatrix} A_{11} & A_{1\Gamma_1} & 0 & 0 \\ 0 & I & 0 & -I \\ 0 & 0 & A_{22} & A_{2\Gamma_2} \\ 0 & 0 & A_{\Gamma_2 2} & A_{\Gamma_2\Gamma_2} \end{pmatrix} \begin{pmatrix} u_1 \\ u_{\Gamma_1} \\ u_2 \\ u_{\Gamma_2} \end{pmatrix} = \begin{pmatrix} b_1 \\ 0 \\ b_2 \\ b_{\Gamma_2} + r_{\Gamma_1} \end{pmatrix}$$

← Dirichlet

← Neumann

$$\begin{pmatrix} A_{11} & A_{1\Gamma_1} & 0 & 0 \\ 0 & I & 0 & -I \\ 0 & 0 & A_{22} & A_{2\Gamma_2} \\ 0 & 0 & A_{\Gamma_2 2} & A_{\Gamma_2\Gamma_2} \end{pmatrix} \begin{pmatrix} u_1 \\ u_{\Gamma_1} \\ u_2 \\ u_{\Gamma_2} \end{pmatrix} = \begin{pmatrix} b_1 \\ 0 \\ b_2 \\ b_{\Gamma_2} + r_{\Gamma_1} + D_2(u_{\Gamma_1} - u_{\Gamma_2}) \end{pmatrix}$$

← Dirichlet

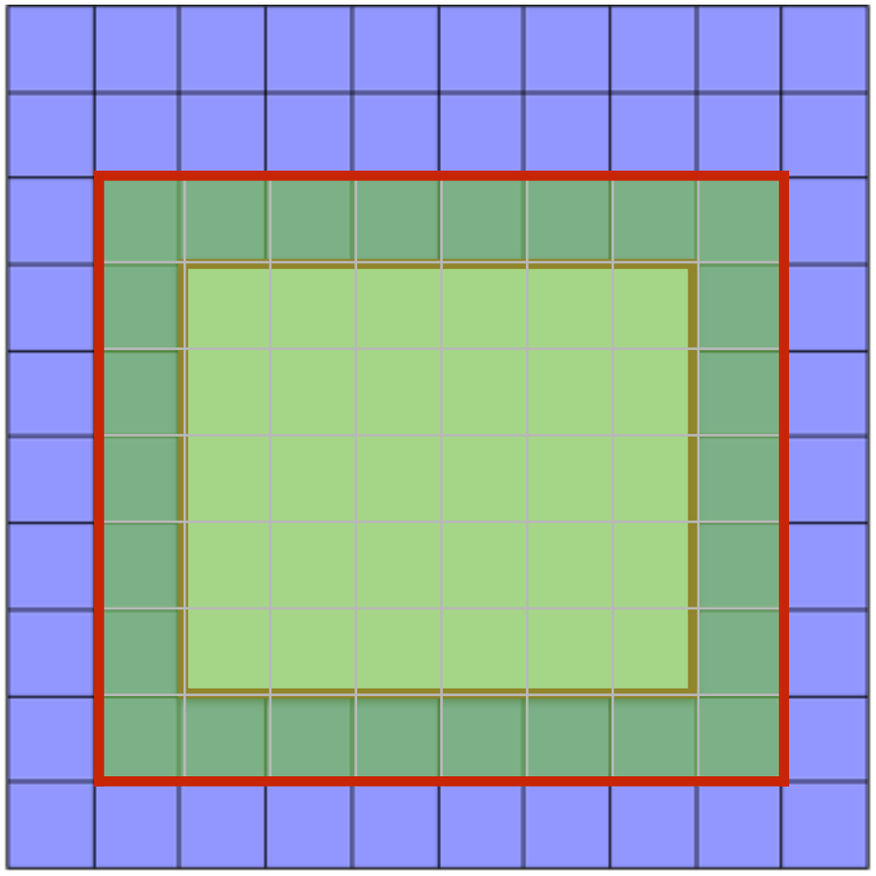
← Robin

$$\begin{pmatrix} A_{11} & A_{1\Gamma_1} & 0 & 0 \\ A_{\Gamma_1 1} & A_{\Gamma_1\Gamma_1} & 0 & 0 \\ 0 & 0 & A_{22} & A_{2\Gamma_2} \\ 0 & 0 & A_{\Gamma_2 2} & A_{\Gamma_2\Gamma_2} \end{pmatrix} \begin{pmatrix} u_1 \\ u_{\Gamma_1} \\ u_2 \\ u_{\Gamma_2} \end{pmatrix} = \begin{pmatrix} b_1 \\ b_{\Gamma_1} + r_{\Gamma_2} + D_1(u_{\Gamma_2} - u_{\Gamma_1}) \\ b_2 \\ b_{\Gamma_2} + r_{\Gamma_1} + D_2(u_{\Gamma_1} - u_{\Gamma_2}) \end{pmatrix}$$

← Robin

← Robin

How? *ALGEBRAIC:* matrix-vector product level



Overlapping

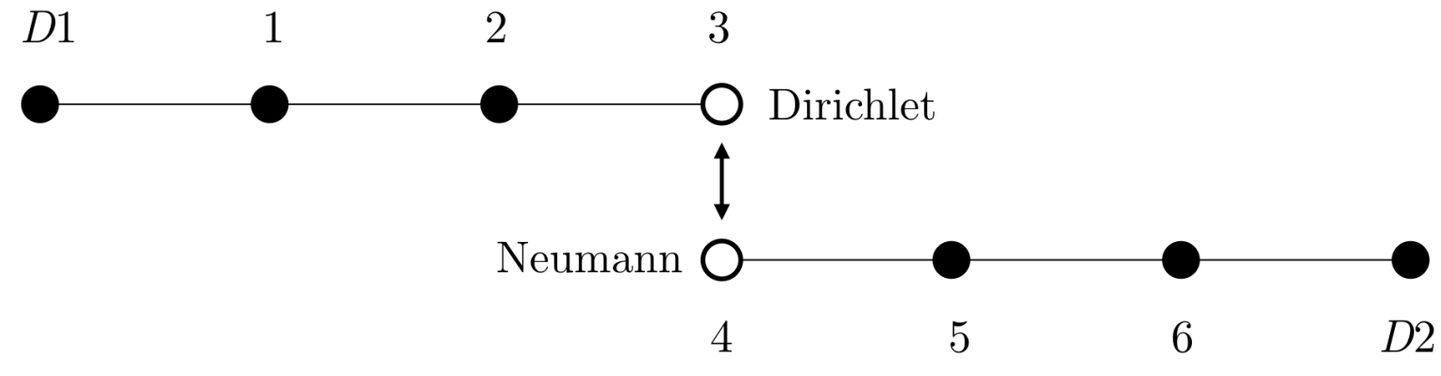
$$\begin{pmatrix} A_{11} & A_{1\Gamma_1} & 0 & 0 \\ 0 & I & 0 & 0 \\ 0 & 0 & A_{22} & A_{2\Gamma_2} \\ 0 & 0 & 0 & I \end{pmatrix} \begin{pmatrix} u_1 \\ u_{\Gamma_1} \\ u_2 \\ u_{\Gamma_2} \end{pmatrix} = \begin{pmatrix} b_1 \\ u_2|_{\Gamma_1} \\ u_2 \\ u_1|_{\Gamma_2} \end{pmatrix}$$

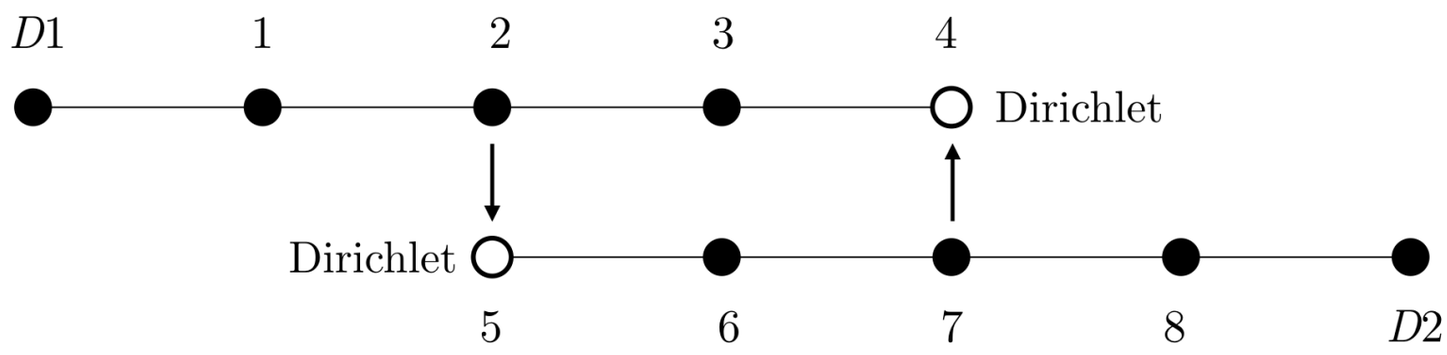
← Dirichlet

← Dirichlet

How? *ALGEBRAIC:* matrix-vector product level

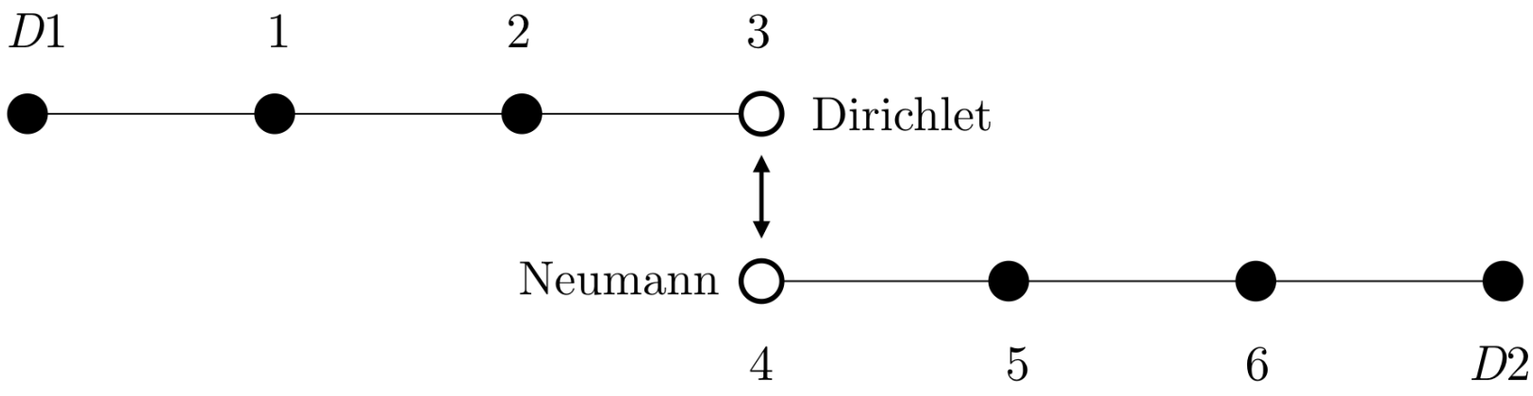
1-dimensional example





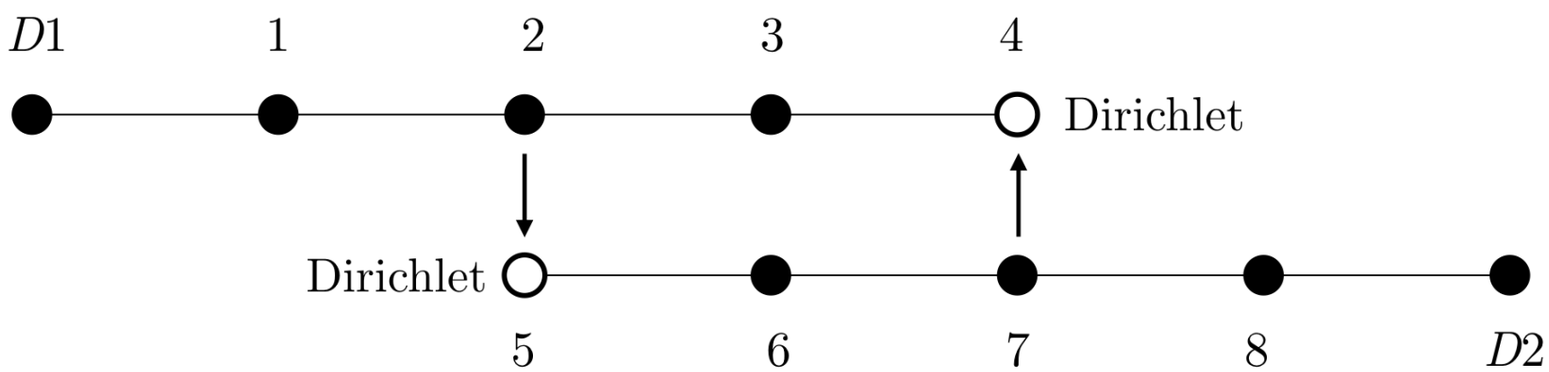
$$-\frac{d^2u}{dx^2} = 0 \text{ on } [0,6] \text{ with } u = x \text{ at } x = 0, 6$$

How? *ALGEBRAIC:* matrix-vector product level



$$\left(\begin{array}{ccc|ccc} 2 & -1 & 0 & & & \\ -1 & 2 & -1 & & \mathbf{0} & \\ 0 & -1 & 1 & & & \\ \hline & & & 1 & -1 & 0 \\ & \mathbf{0} & & -1 & 2 & -1 \\ & & & 0 & -1 & 2 \end{array}\right) \left(\begin{array}{c} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{array}\right) = \left(\begin{array}{c} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 6 \end{array}\right)$$

How? *ALGEBRAIC:* matrix-vector product level



$$
\left(\begin{array}{cccc|cccc}
2 & -1 & 0 & 0 & & & & \\
-1 & 2 & -1 & 0 & & & & \\
0 & -1 & 2 & -1 & & \mathbf{0} & & \\
0 & 0 & -1 & 1 & & & & \\
\hline
 & & & & -1 & 1 & 0 & 0 \\
 & & & & -1 & 2 & -1 & 0 \\
 & \mathbf{0} & & & 0 & -1 & 2 & -1 \\
 & & & & 0 & 0 & -1 & 2
\end{array}\right)
\left(\begin{array}{c} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_8 \end{array}\right)
=
\left(\begin{array}{c} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 6 \end{array}\right)
$$

How? *ALGEBRAIC*: matrix-vector product level

Algorithm Implicit Dirichlet/Neumann and Dirichlet/Dirichlet applied to the preconditioned Richardson method

1: Initial condition: $k = 0$, $\mathbf{u}^0$, equalize $\mathbf{u}^0$ on interface and overlapping zone
2: Diagonal: $\mathbf{d} = \text{diag}(\mathbf{A})$
3: Apply Domain Composition Method on $\mathbf{d}$ and $\mathbf{b}$:

$$\text{D/N:} \begin{cases} d_4 = d_4 + d_3 & \text{then } d_3 = d_4 \\ b_4 = b_4 + b_3 & \text{then } b_3 = b_4 \end{cases}$$

$$\text{D/D:} \begin{cases} d_4 = d_7, & d_5 = d_2 \\ b_4 = b_7, & b_5 = b_2 \end{cases}$$

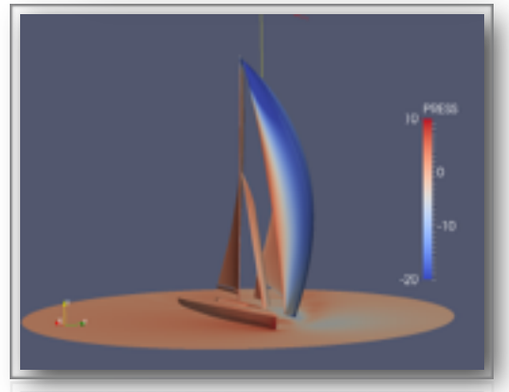
4: **while** Convergence not achieved **do**
5: Compute: $\mathbf{q} = \mathbf{A}\mathbf{u}^k$
6: Apply Domain Composition Method on $\mathbf{q}$:

$$\text{D/N:} \begin{cases} q_4 = q_4 + q_3 & \text{then } q_3 = q_4 \end{cases}$$

$$\text{D/D:} \begin{cases} q_4 = q_7, & q_5 = q_2 \end{cases}$$

7: Residual: $\mathbf{r}^k = \mathbf{b} - \mathbf{q}$
8: Solution update: $\mathbf{u}^{k+1} = \mathbf{u}^k + \mathbf{d}^{-1} \cdot \mathbf{r}^k$
9: $k = k + 1$
10: **end while**

How? *ALGEBRAIC:* matrix-vector product level



Multi-physics modular code for High Performance Computational Mechanics & Design

Born in 2004

Alya

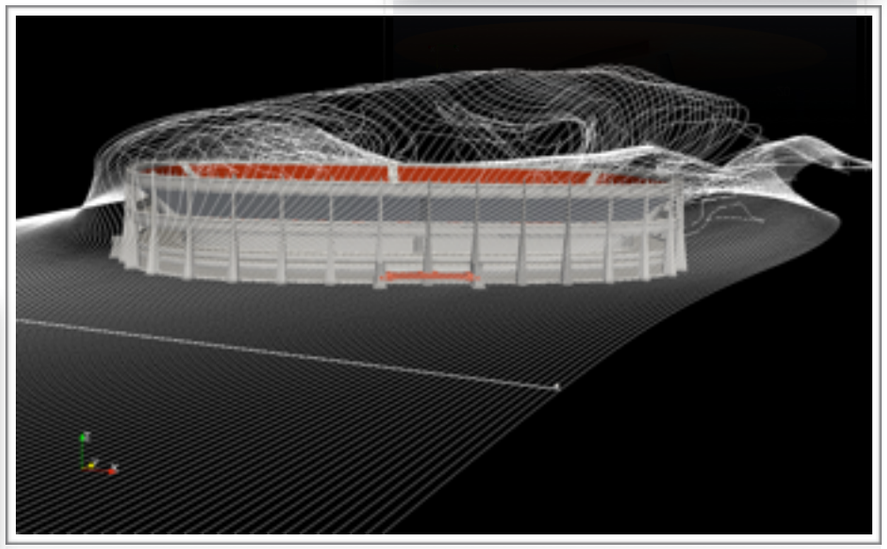
Numerical solution of PDE's

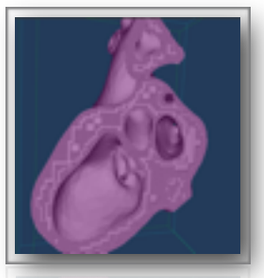
Variational methods are preferred (FEM)

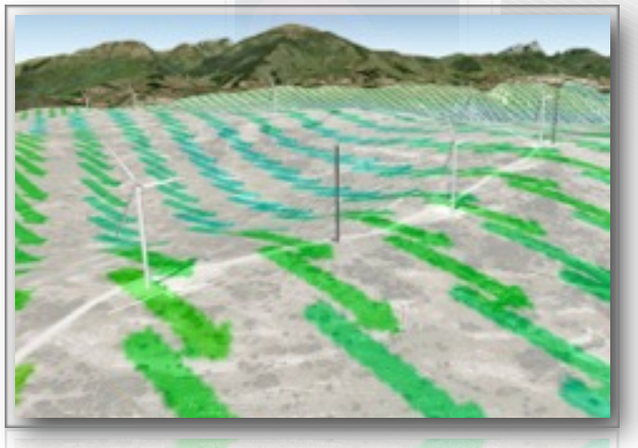
Coupling between multi-physics (loose or strong)

Explicit and Implicit formulations

Hybrid meshes, non-conforming meshes





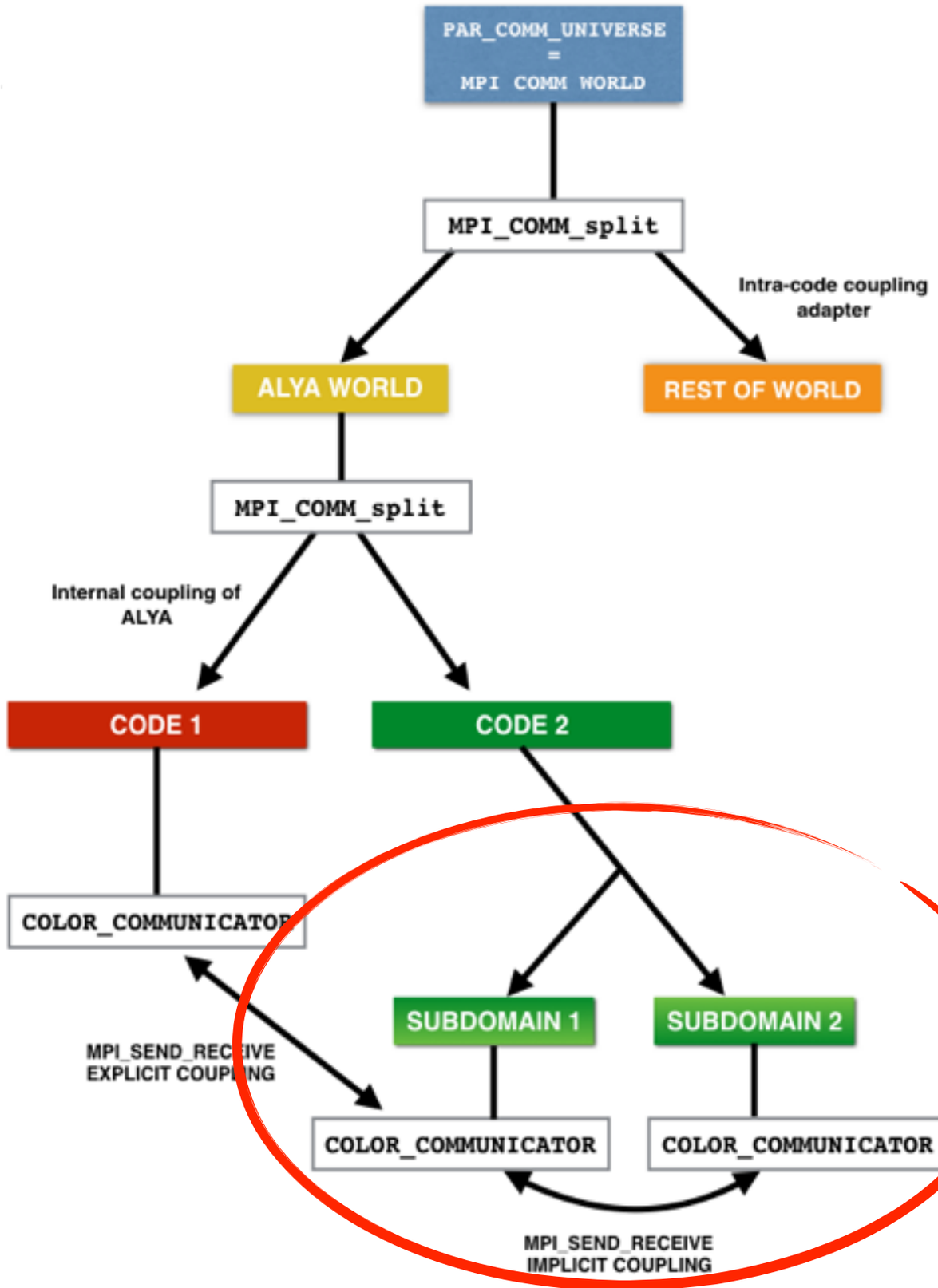


Parallelization by MPI and OpenMP

Automatic mesh partition using Metis

Portability is a must

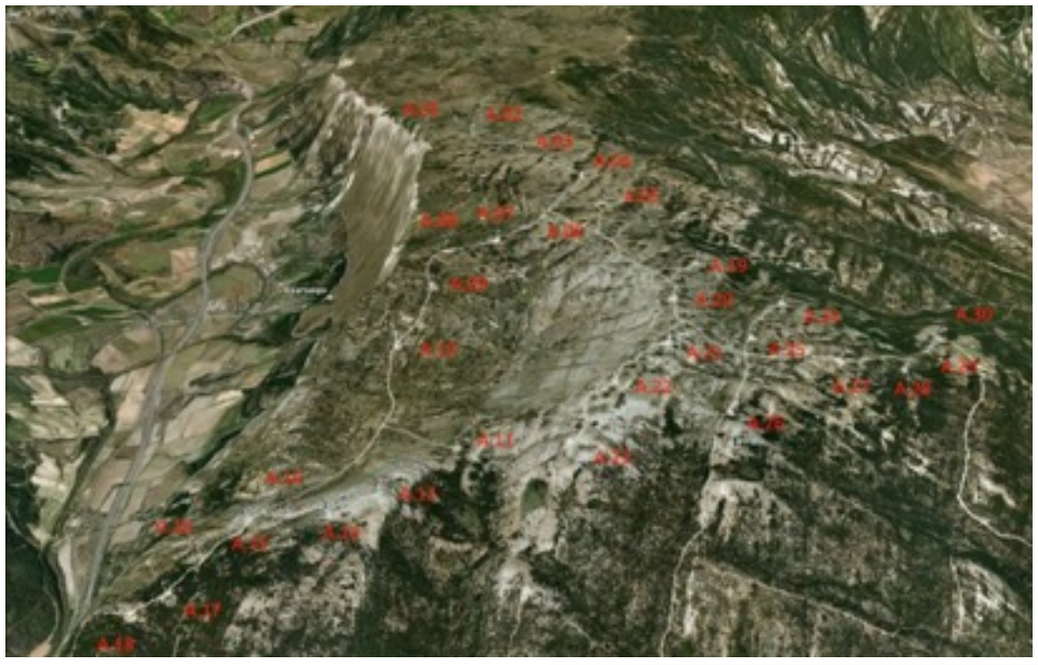
Where?

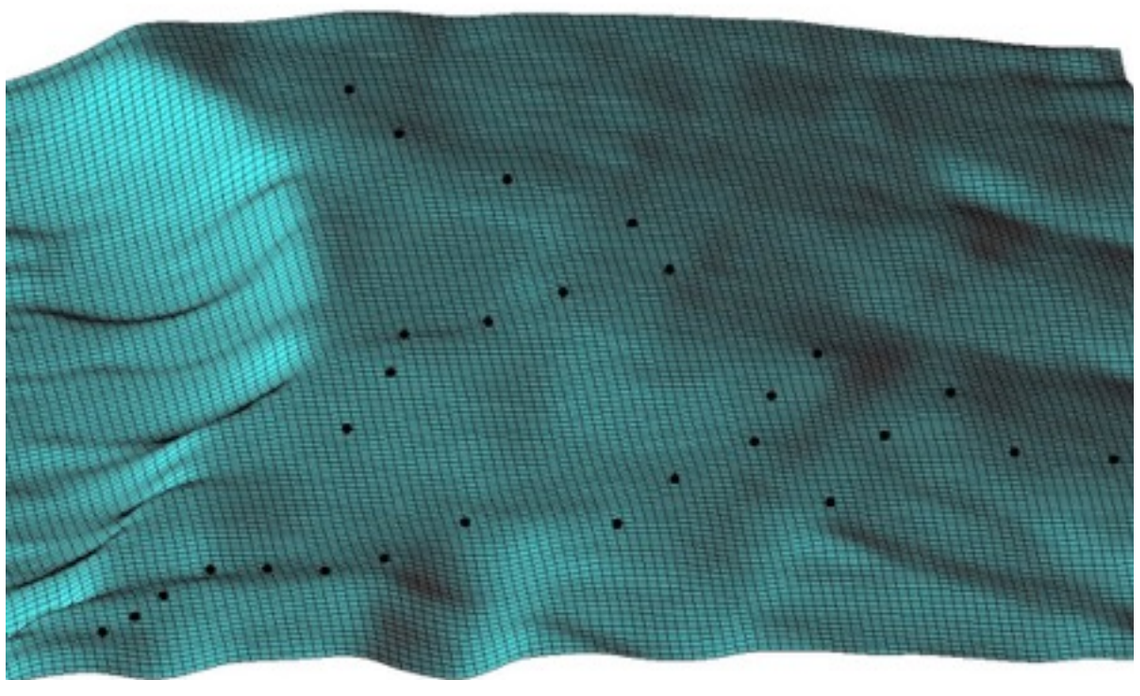

PAR_COMM_UNIVERSE
=
MPI COMM WORLD
MPI_COMM_split
Intra-code coupling
adapter
ALYA WORLD
REST OF WORLD
MPI_COMM_split
Internal coupling of
ALYA
CODE 1
CODE 2
COLOR_COMMUNICATOR
SUBDOMAIN 1
SUBDOMAIN 2
MPI_SEND_RECEIVE
EXPLICIT COUPLING
COLOR_COMMUNICATOR
COLOR_COMMUNICATOR
MPI_SEND_RECEIVE
IMPLICIT COUPLING


my_universe_rank
my_world_rank
my_code_rank
my_color_rank

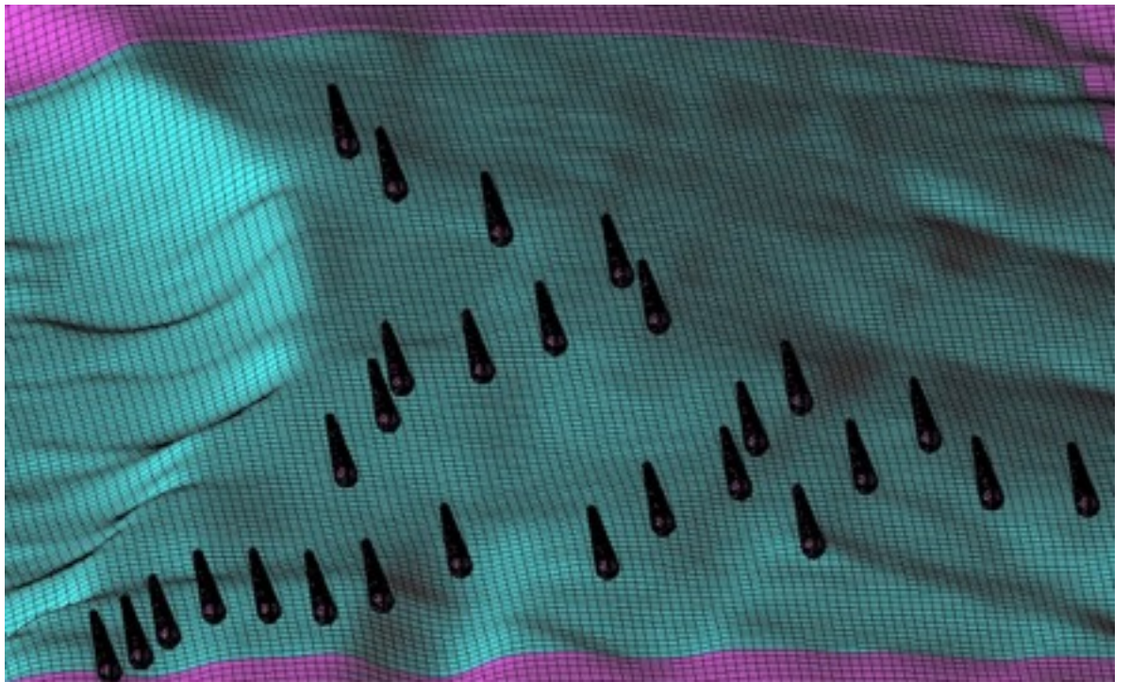
Real problem

Actuator disk simulation



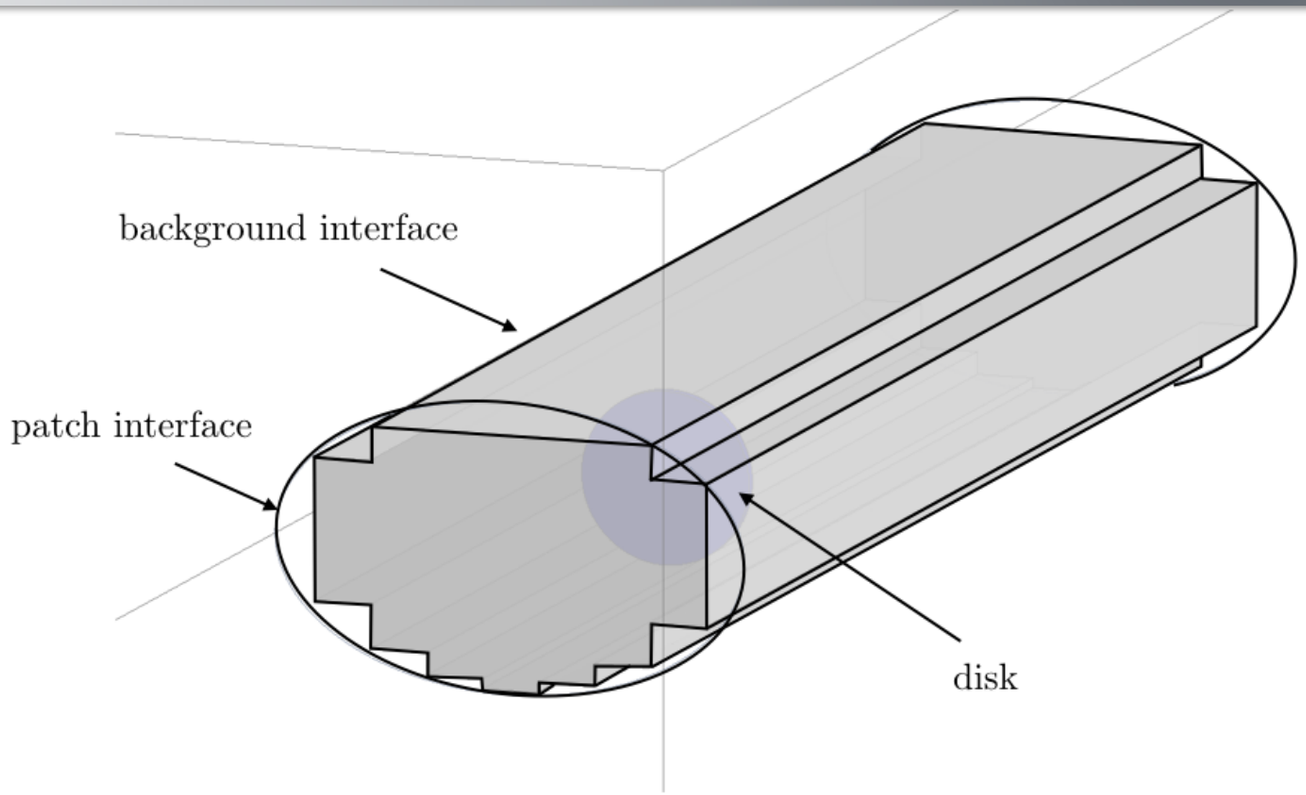


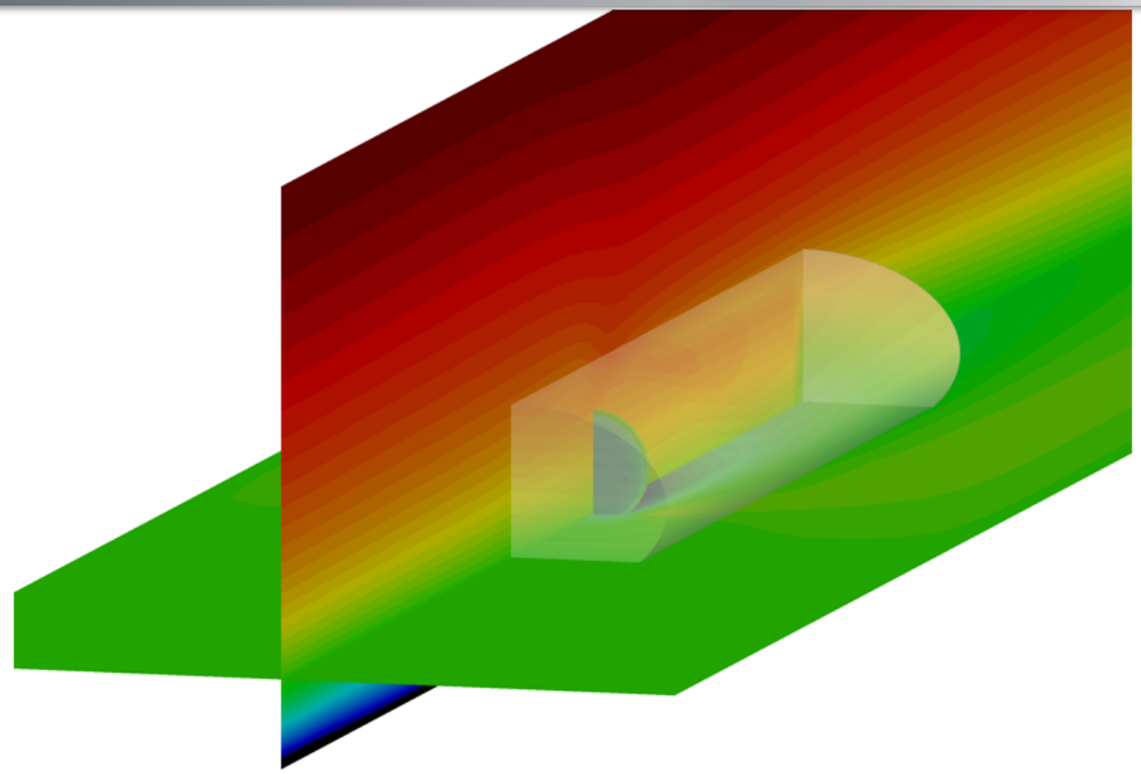
Disks in the position of the turbines

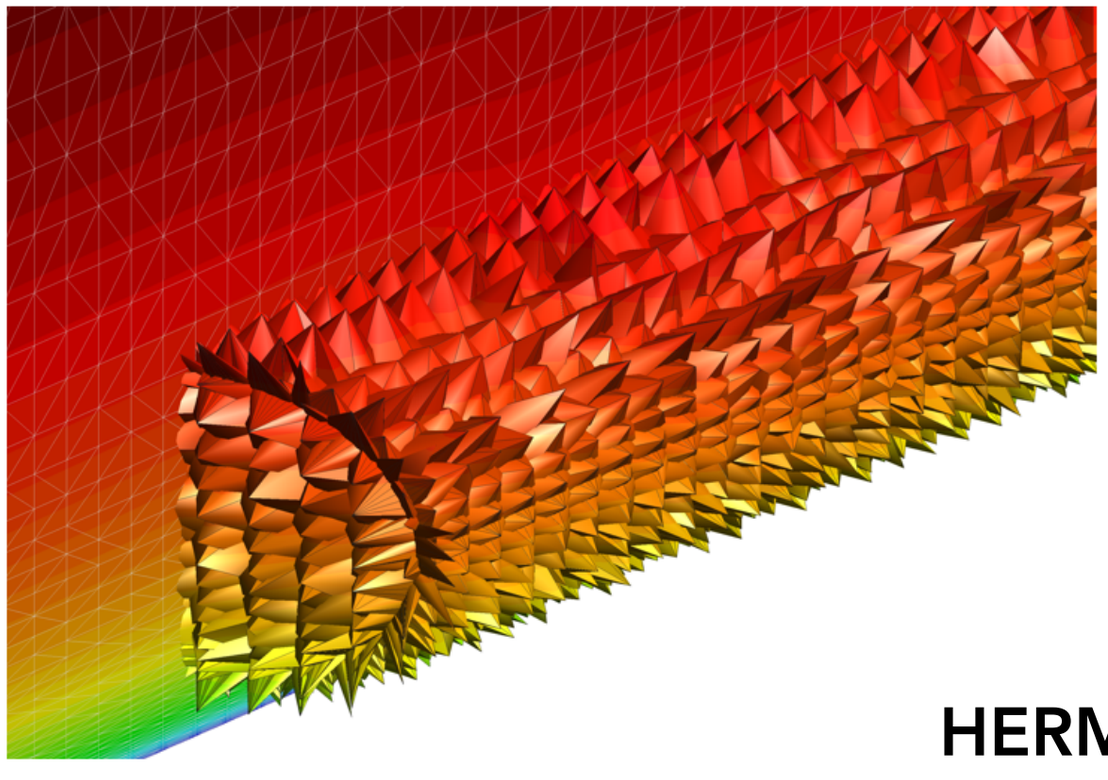


Patch containing the disks

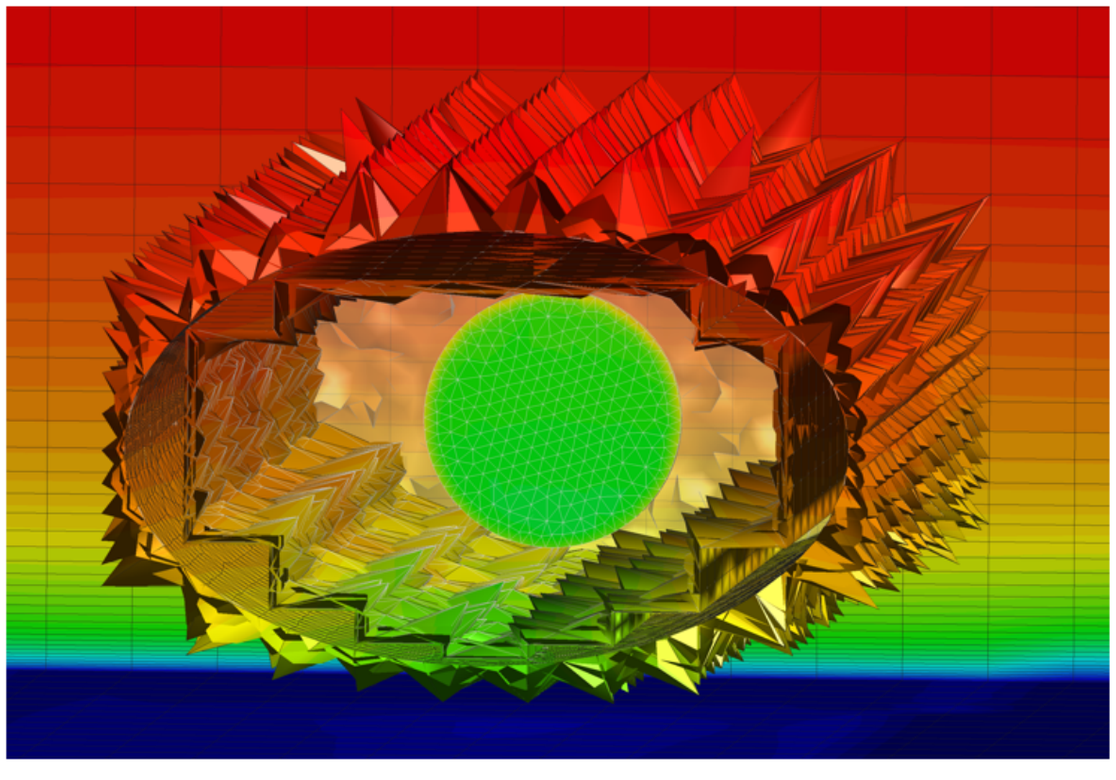
Real problem



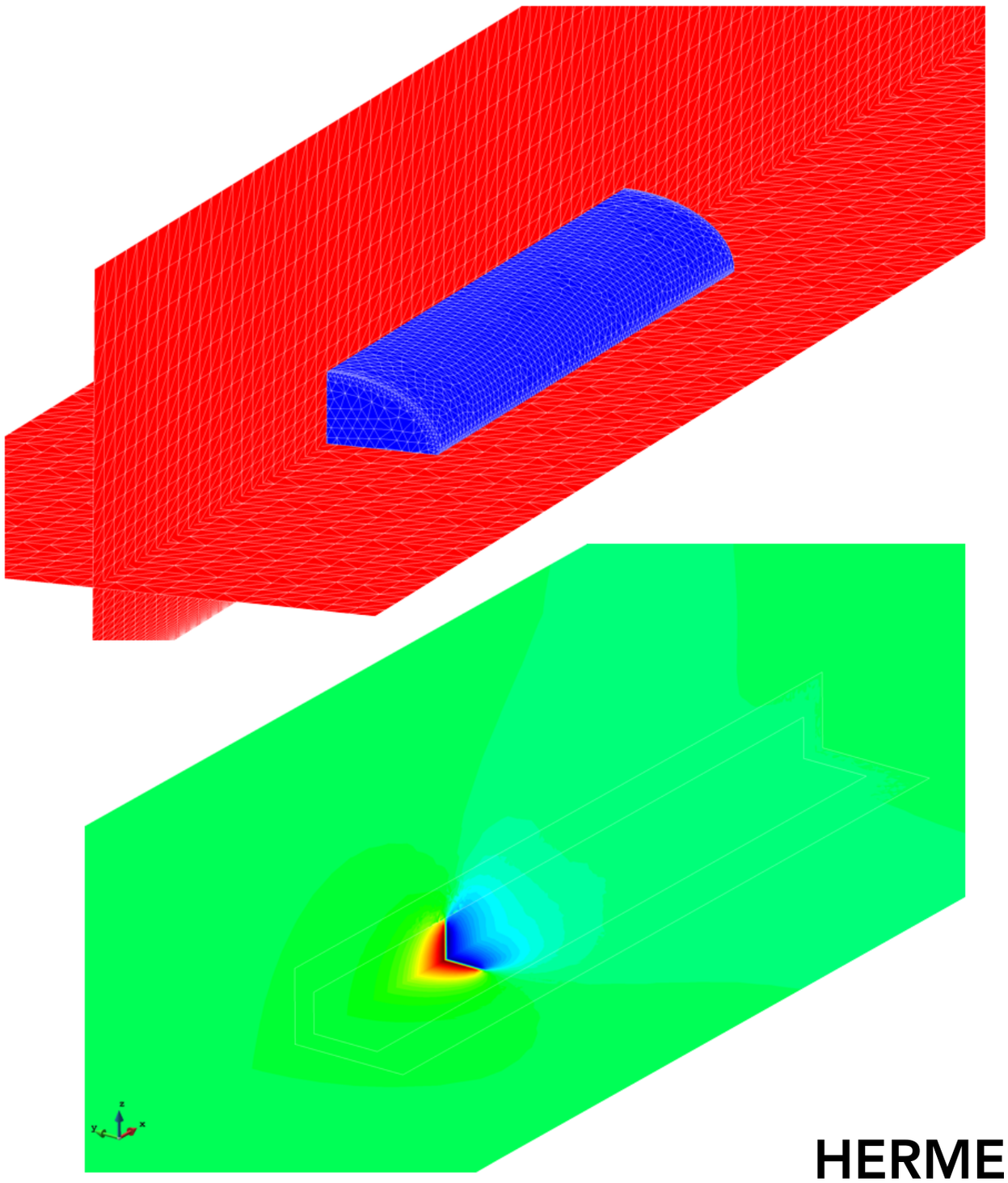


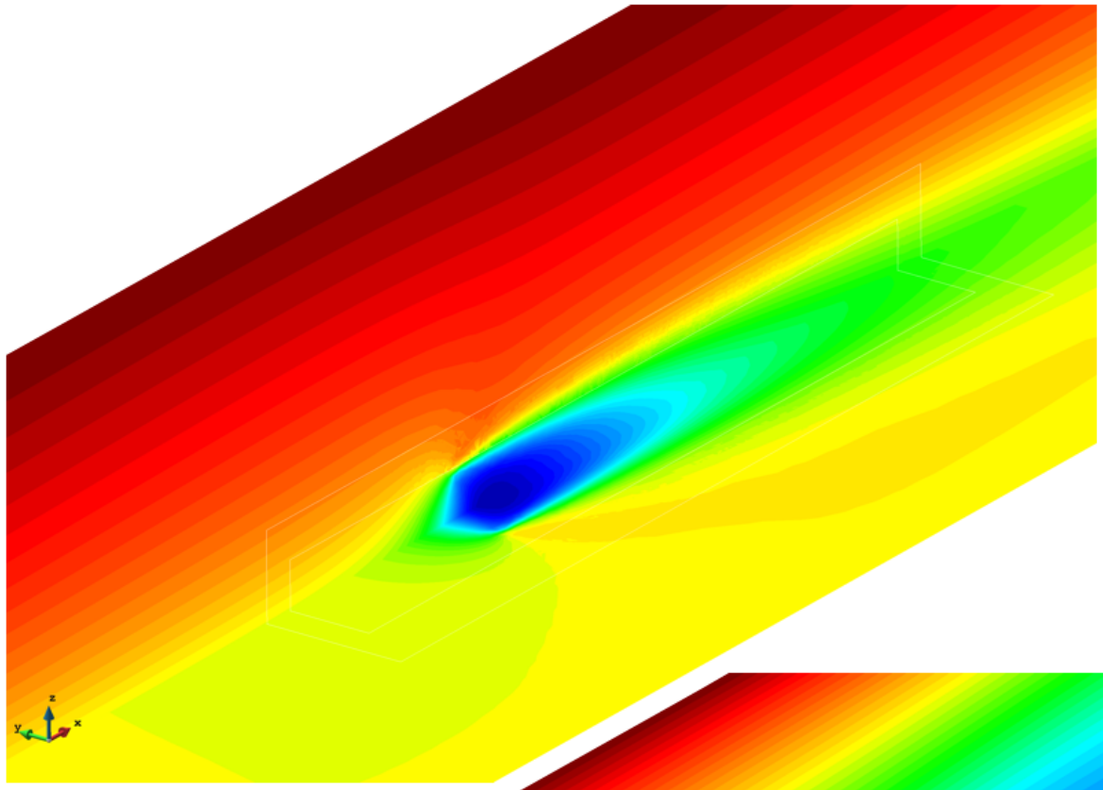


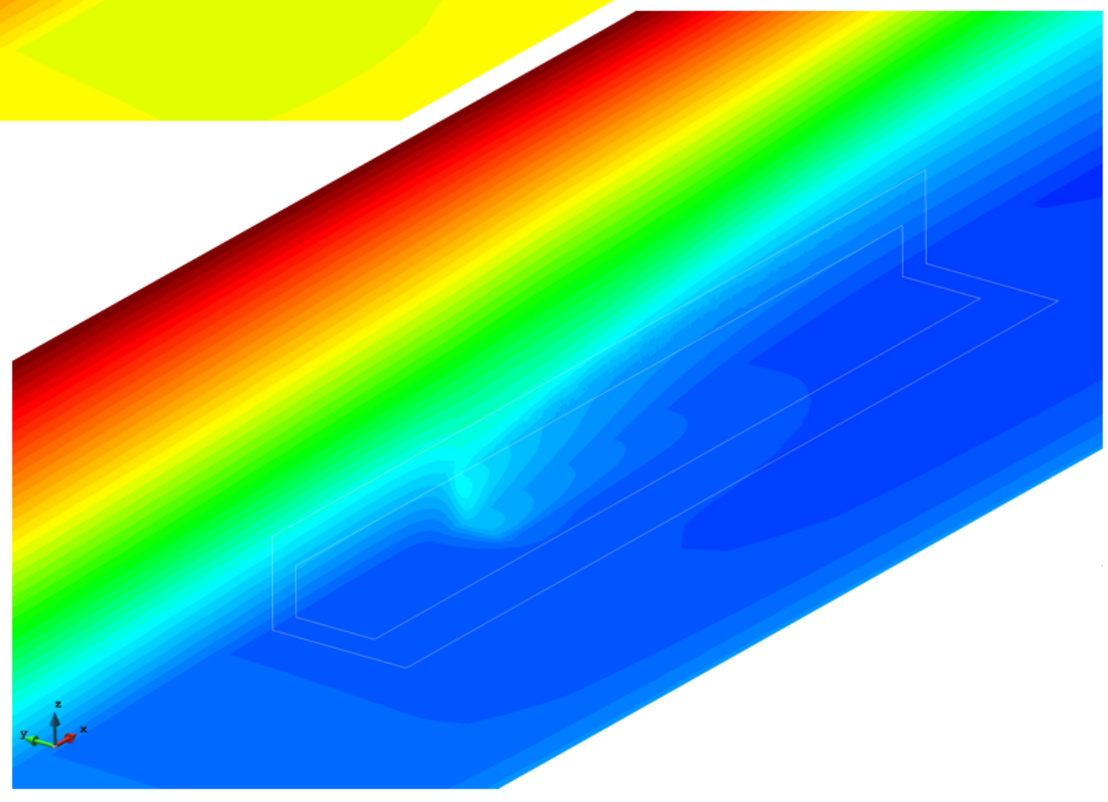
HERMESH



Real problem

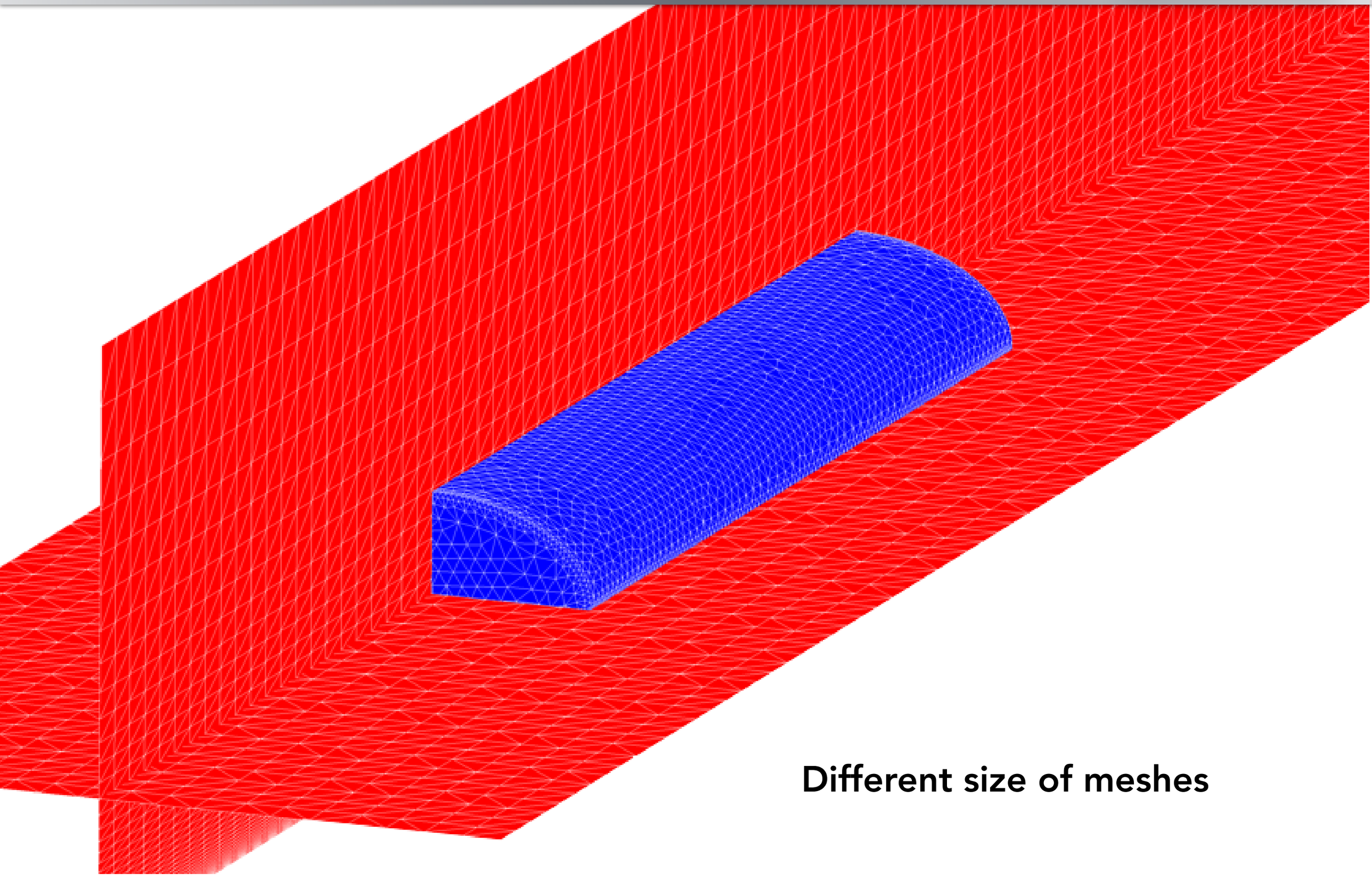






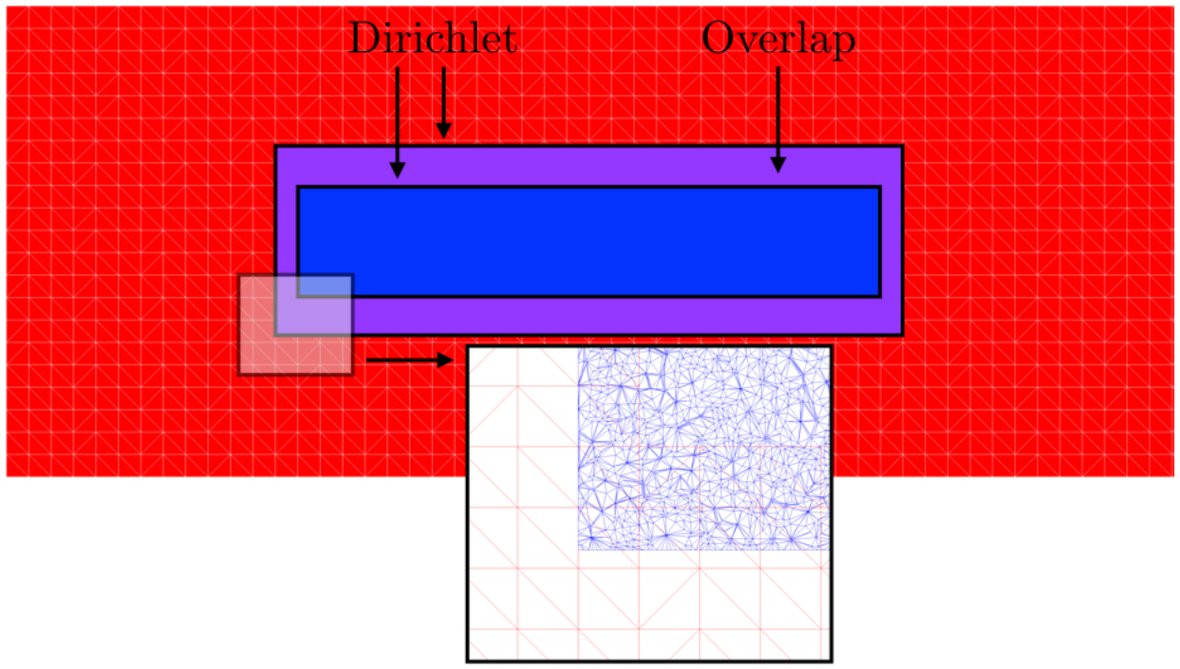
HERMESH

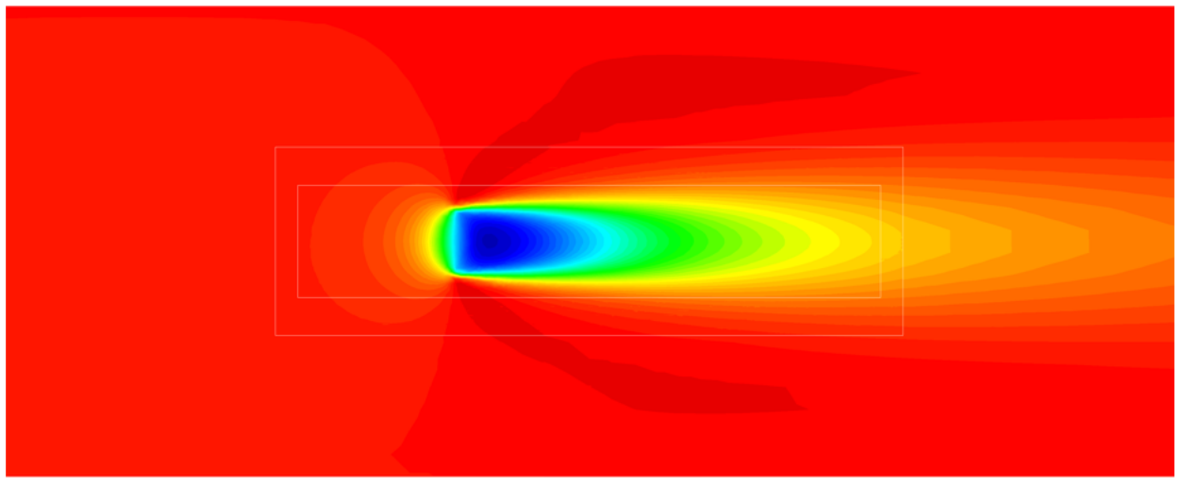
Real problem

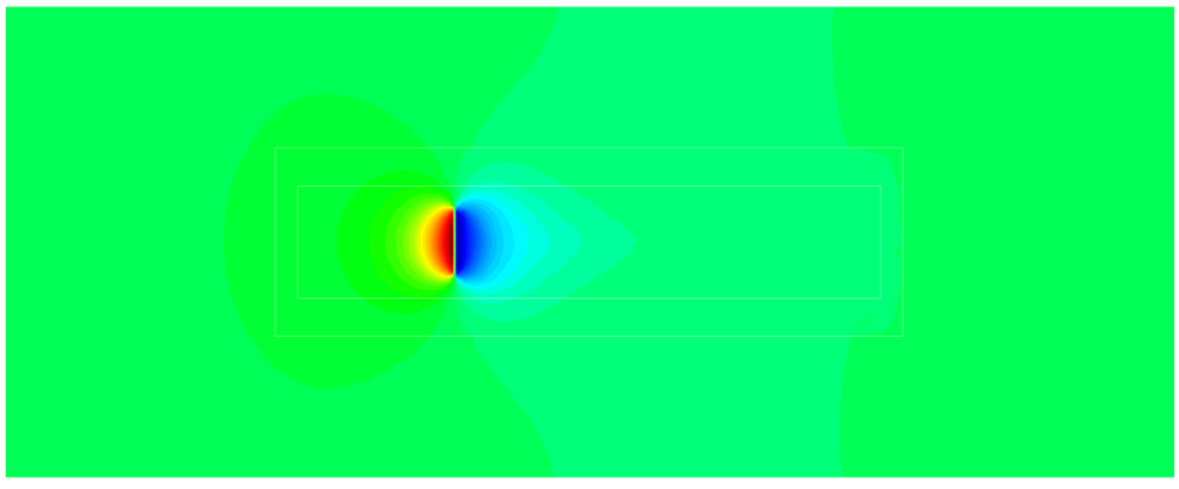


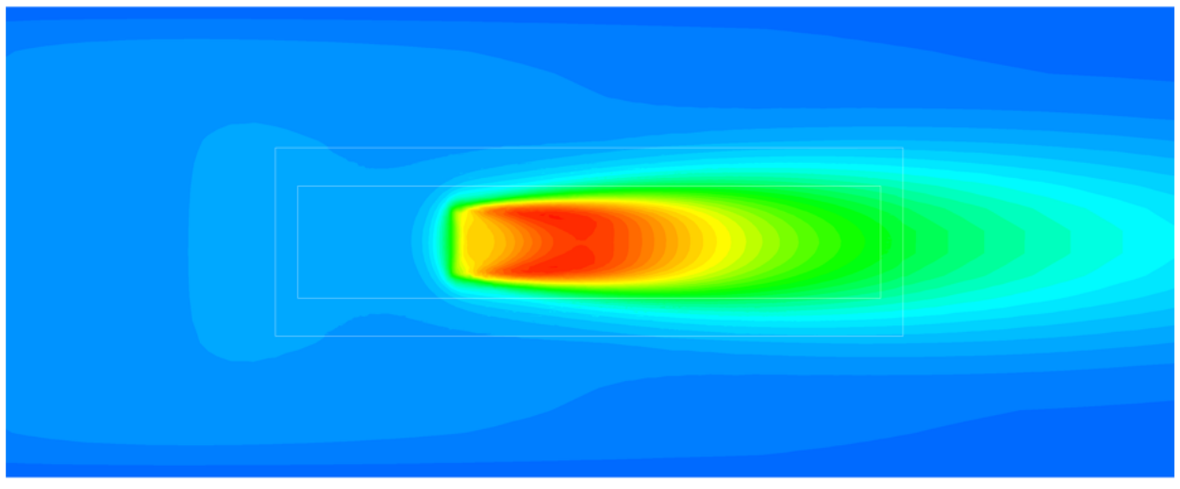
Different size of meshes

Real problem



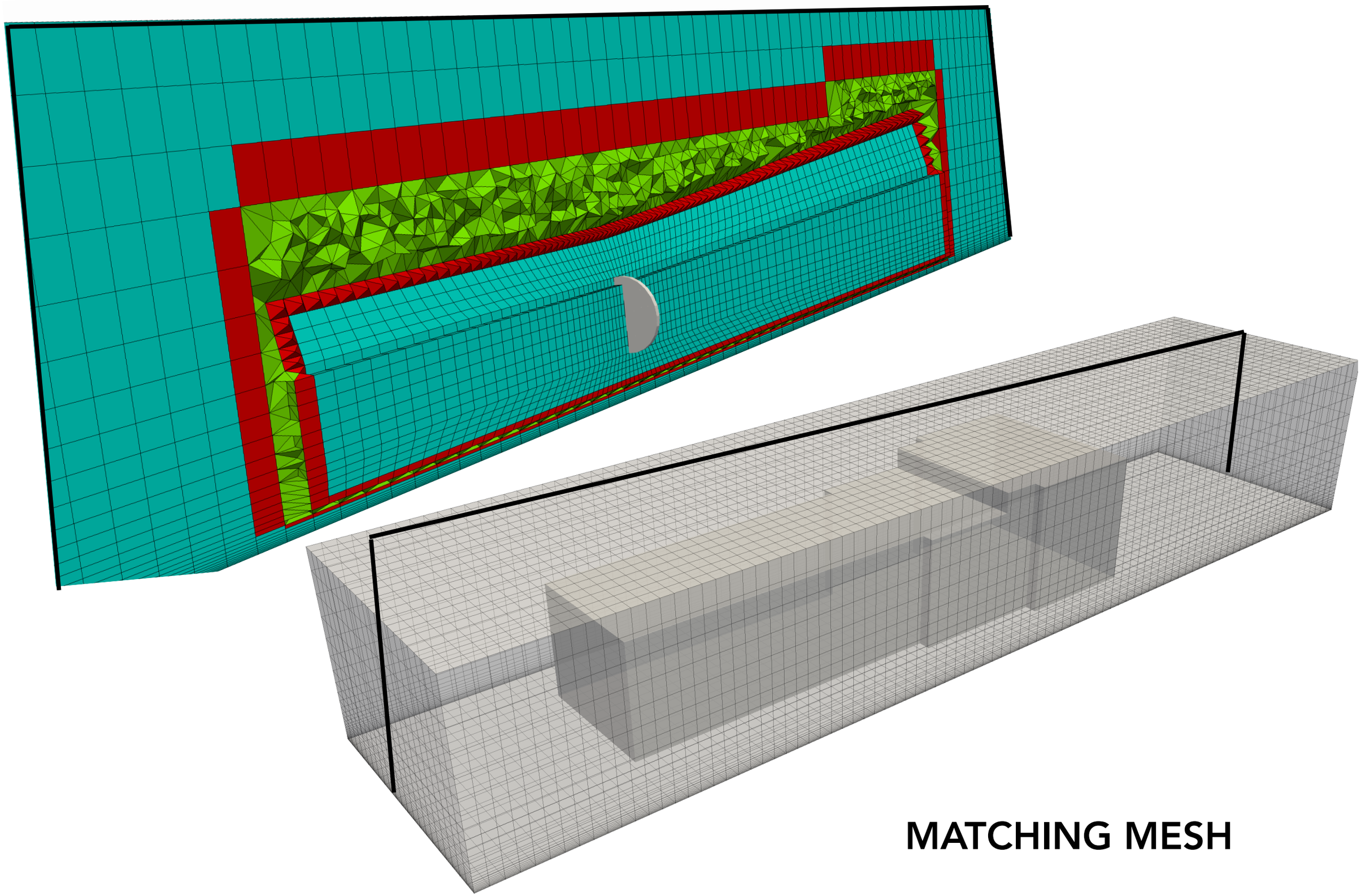






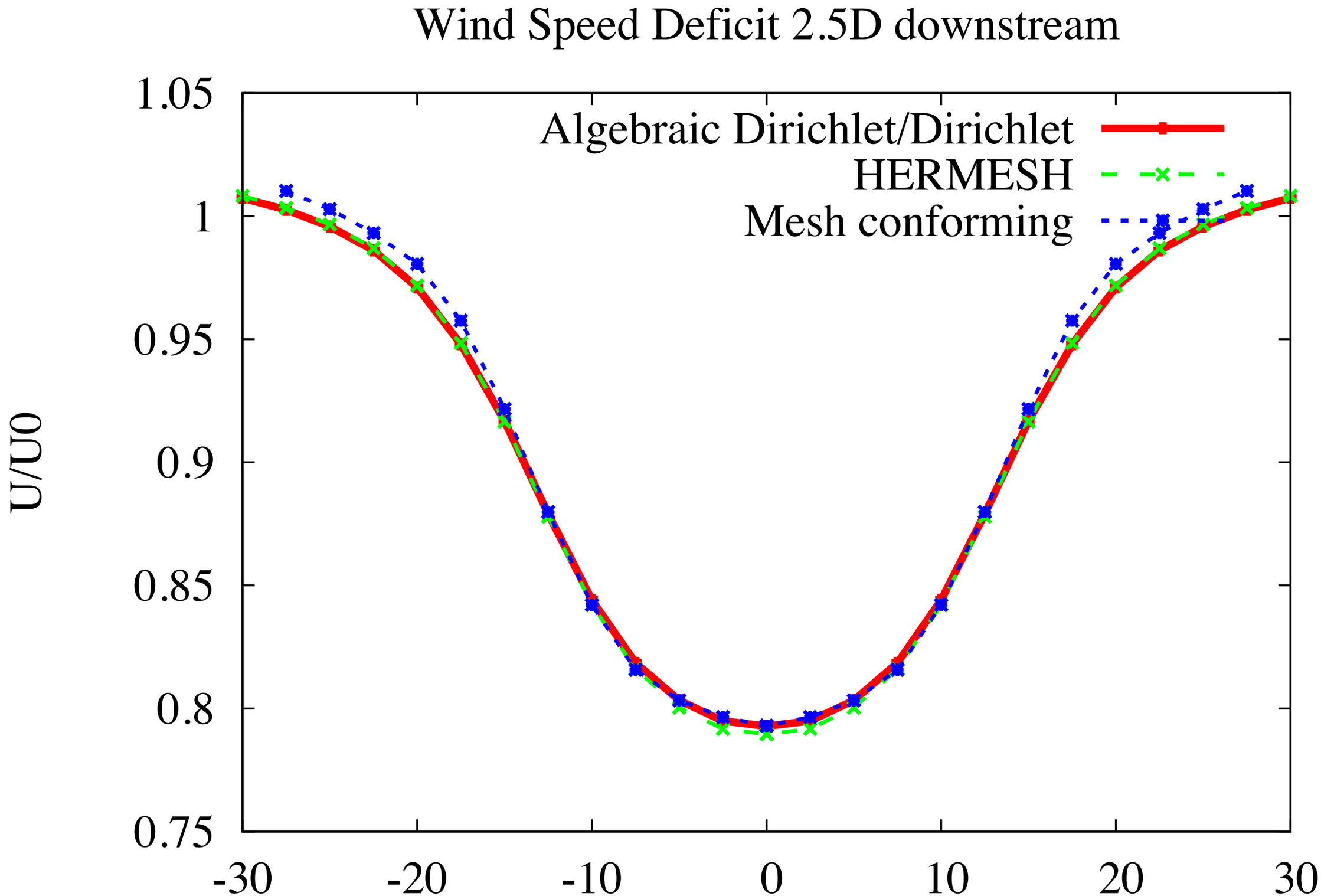
ALGEBRAIC

Real problem



MATCHING MESH

Real problem



GRACIAS!